Lung Cancer Subtyping from Gene Expression Data using General and Enhanced Fuzzy Min-Max Neural Networks

Yashpal Singh[1] and Seba Susan[1]

[1]Delhi Technological University

December 7, 2022

Abstract

Cancer diagnosis using gene expression data is significant research for facilitating early treatment and prevention of cancer. The classification of gene expression data is challenging due to its high dimensionality and smaller number of samples that renders classification a difficult task. Creation of well-defined class boundaries is the aim of every classification algorithm. The Fuzzy min-max (FMM) neural network classifier is known to create good decision boundaries using hyperboxes constructed for each class. In this paper, we explore the General Fuzzy min-max (GFMM) and Enhanced Fuzzy min-max (EFMM) neural network architectures for the classification of lung cancer subtypes from microarray gene expression data. Both GFMM and EFMM are advanced versions of Simpson's FMM neural network classifier. The GFMM is extremely efficient because it involves very simple operations for hyperbox manipulation, and can handle both labeled and unlabeled data. On the other hand, EFMM proposes three heuristic rules related to hyperbox expansion, contraction and the overlap test, which enhances the learning algorithm. We perform the classification of gene expression data using these two algorithms, then we analyze the performance by visualizing the hyperboxes obtained after training, and compare the accuracies of these classifiers. LASSO is used for selecting the important genes from the high-dimensional gene expression data. After the analysis of the results, we observe that EFMM with LASSO gives the best performance as compared to GFMM, FMM and other machine learning algorithms.

LUNG CANCER SUBTYPING FROM GENE EXPRESSION DATA USING GENERAL AND ENHANCED FUZZY MIN-MAX NEURAL NETWORKS

Yashpal Singh and Seba Susan*

Department of Information Technology,
Delhi Technological University, Delhi, India
*seba_406@yahoo.in

Abstract. Cancer diagnosis using gene expression data is significant research for facilitating early treatment and prevention of cancer. The classification of gene expression data is challenging due to its high dimensionality and smaller number of samples that renders classification a difficult task. Creation of well-defined class boundaries is the aim of every classification algorithm. The Fuzzy min-max (FMM) neural network classifier is known to create good decision boundaries using hyperboxes constructed for each class. In this paper, we explore the General Fuzzy min-max (GFMM) and Enhanced Fuzzy min-max (EFMM) neural network architectures for the classification of lung cancer subtypes from microarray gene expression data. Both GFMM and EFMM are advanced versions of Simpson's FMM neural network classifier. The GFMM is extremely efficient because it involves very simple operations for hyperbox manipulation, and can handle both labeled and unlabeled data. On the other hand, EFMM proposes three heuristic rules related to hyperbox expansion, contraction and the overlap test, which enhances the learning algorithm. We perform the classification of gene expression data using these two algorithms, then we analyze the performance by visualizing the hyperboxes obtained after training, and compare the accuracies of these classifiers. LASSO is used for selecting the important genes from the high-dimensional gene expression data. After the analysis of the results, we observe that EFMM with LASSO gives the best performance as compared to GFMM, FMM and other machine learning algorithms.

Keywords: Gene expression data, fuzzy min-max neural network, LASSO.

1 Introduction

Cancer diagnosis using gene expression signatures is trending research due to its usefulness in early treatment and detection of cancer which is a major cause of death worldwide. Mining of gene expression data has also attracted data mining researchers due to the numerous challenges involved that makes it distinct from patterns found in normal data. Microarray gene expression data is very challenging to work with due to its high dimensionality and limited number of samples. The problem is complicated by the presence of noise, and an imbalanced class distribution in which one type or subtype of cancer has a larger population than other classes. In literature, a large variety of machine

learning algorithms such as neural networks, support vector machines, logistic regression, gradient boosting machines, naïve bayes, and random forest of decision trees have been applied for the classification of gene expression data. Slonim et al. [8] distinguished between class discovery and class prediction for gene expression data in a Bayesian inferencing framework; they used leukemia dataset, and they found that the genes without correlation give better results, and the median prediction was 0.86. Khan et al. [9] used Artificial Neural Networks (ANNs) for the categorization of cancer using gene expression profiles; the main advantage found was that it could work with non-linear features and has high sensitivity. Lyu et al. [10] performed tumor classification using a Convolution Neural Network (CNN) for learning from the gene expression data, and they achieved 95.59% of accuracy which was better as compared to other related works. For the classification of gene expression profiles, Ahmed et al. [11] used the Deep Neural Network (DNN), improved DNN, CNN, and RNN along with preprocessing techniques; improved DNN gave the best result of all.

A host of classifiers such as the support vector machine [12], random forest of decision trees [13], logistic regression [14] and naïve Bayes classifier [15] have been successfully used for the classification of gene expression data. All these classifiers work on crisp data without transiting to the fuzzy domain. Feature selection techniques like LASSO has proved to improve the performance of these classifiers on gene expression data [16]. The combination of fuzzy sets and classification is well covered in literature [17-19]. The motivation behind using fuzzy classifiers is the creation of fuzzy decision boundaries in the input space, which allows for flexible decision making. The FMM, GFMM, EFMM neural networks are examples of successful application of the fuzzy set theory to pattern recognition and classification problems. A brief introduction of the same is introduced here. As we know, regarding crisp sets, the element or a data point that is in the universe of discourse either belongs to the positive set (i.e. 1) or not (i.e. 0). But the fuzzy sets are more generalized and regard all samples as members of a set; they take care of the data points which partially belong to a set by calculating their membership values with respect to the particular set. The membership value is continuous between [0, 1], and it is different for each data point. By using this concept, Simpson *et al.* [1] introduced the Fuzzy min-max (FMM) neural network in 1992 which has been applied successfully for classification and clustering problems. In FMM, the fuzzy set is represented by a rectangle which is also known as hyperbox. All hyperboxes have min and max points, and the data point which lies inside the hyperbox has a membership value equal to 1. The fuzzy min-max classifier connects the hyperboxes to their respective class nodes (i.e. gives the highest membership value) which can be used for the classification. The learning algorithm is divided into three phases, first, it checks for the hyperbox expansion then after the successful expansion it goes for the second phase which performs the overlap test. If there is overlap, then the final phase of the learning algorithm is contraction phase which removes the unwanted overlap of hyperboxes. Many researchers modify this learning algorithm to make it more efficient and fast [2]. The two popular improved versions of the FMM classifier which we are going to apply in this paper are the General Fuzzy Min-Max (GFMM) [3] and Enhanced Fuzzy Min-Max (EFMM) [4] neural networks. The GFMM improves the effectiveness of the original fuzzy min-max algorithm by suggesting a few modifications to the general FMM architecture and functioning, some of which are given below.

1. In the pattern space, the input patterns can be fuzzy hyperbox or crisp points.
2. The membership function and the hyperbox expansion constraints are modified.
3. This algorithm can be used for both clustering and classification because it can process labeled and unlabeled input points at the same time.
4. In the original algorithm, the number of hyperboxes created depends on the maximum hyperbox size hyperparameter θ. The smaller the value θ, the more the number of hyperboxes created, and this leads to the overfitting problem. Larger θ creates lesser number of hyperboxes which increases the generalization ability, but the ability to capture the boundaries between the classes is decreased. So the settlement between these two cases is implemented in generalized fuzzy min-max.

In the original fuzzy min-max, Simpson proposed two different algorithms for classification and clustering problems but the GFMM combines them in one algorithm. The training of GFMM is extremely efficient for almost every case because it uses very simple compare, add and subtract operations for the hyperbox manipulation.

The other very popular version of FMM is Enhanced fuzzy min-max (EFMM) [4] which is known to give high classification performance. There are three heuristic rules introduced in the EFMM which enhances the learning algorithm. First, reducing the overlapping regions of hyperbox during the expansion phase reduces the classification errors. Second, the already existing overlap testing phase is extended so all the overlapping corners can be identified. Third, the existing hyperbox contraction rule in FMM is not able to cover all the overlapping cases, so in EFMM they introduced a new rule for contraction for solving the different overlapping cases.

In this paper, we investigate the application of GFMM and EFMM classifiers for classification of lung cancer gene expression data. The application of GFMM and EFMM to gene expression data has not been explored before. In a recent work [5], authors applied the FMM classifier for the classification of lung cancer gene expression data. The current work advances on this work by exploring two advanced versions of the FMM classifier for the application on microarray data. The aim is to exploit the improved definitions of hyperboxes and expansion-contraction learning process for determining the decision boundaries between cancer subtypes.

The problem with the microarray data is they have thousands of genes (or features), and processing of all the features takes a larger time as compared to when using fewer features. To overcome this issue, we use Least Absolute Shrinkage and Selection Operator (LASSO) for the feature selection which a high performing algorithm for skewed feature selection [6][7]. Then we calculate the performance of both the fuzzy classifiers and then compare the results with that of several other machine learning algorithms. We perform fivefold cross-validation for all the classification algorithms, and all the comparisons are made based on accuracy and execution time of the algorithm.

The organization of this paper is as follow. Sections 2 and 3 contain a brief discussion about GFMM and EFMM, respectively. Section 4 presents the methodology used for the experiments, and finally, section 5 analyzes the classification results, and section 6 outlines the future scope of this work.

2 General Fuzzy Min-Max Neural Network

In this section we discuss about the input patterns of GFMM, learning algorithm phases and the neural network at the core of GFMM for the current task of classification of microarray gene expression data.

2.1 Input pattern

The input pattern that is processed by the GFMM is the ordered pair of the h^{th} input pattern and the class index of one of the classes. The ordered pair is given by

$$\{I_h\ , c_h\} \tag{1}$$

where I_h is the h^{th} input pattern in the form of I_h^l (lower) and I_h^u (upper) i.e. [I_h^l , I_h^u] are the vector inputs.

$c_h \in \{0, 1, 2, 3, \ldots, p\}$ is the class index of any one of the p+1 classes. If $c_h = 0$, it means the input is unlabeled.

2.2 Membership Function

The fuzzy hyperbox membership function plays an important role in deciding whether a particular input belongs to a particular class or not. In GFMM, the new membership function is defined which fulfills the limitations of the original fuzzy min-max. In the original function, it was observed that by increasing the distance from the hyperbox, the membership does not decrease steadily, which is the major drawback of this membership function. In GFMM, the degree of membership I_h for the hyperbox B_q is 1 if I_h is inside the hyperbox B_q, and the membership decreases as the distance from the hyperbox is increases. In the membership equation, $\gamma = [\gamma_1, \gamma_2, \ldots, \gamma_n]$ is the sensitivity parameter; this regulates how fast the membership values decreases.

$$B_q(I_h) = min_{p=1\ to\ n}\left(min\begin{pmatrix}[1 - f(I_{hp}^u - w_{qp}, \gamma_p)],\\ [1 - f(v_{qp} - I_{hp}^l, \gamma_p)]\end{pmatrix}\right) \tag{2}$$

$$where\ , f(r, \gamma) = \begin{Bmatrix} 1 \ \ if\ r\gamma > 1 \\ r\gamma\ \ if\ 0 \le r\gamma \le 1 \\ 0 \ \ if\ r\gamma < 0 \end{Bmatrix} \tag{3}$$

2.3 GFMM Learning Algorithm

The steps of the GFMM learning algorithm are given below.

1. *Min and Max Point Initialization*:
   For the new hyperbox, the algorithm initializes its min point $V_q = 0$ and the max point $W_q = 0$, this can be automatically used in the expansion phase of the algorithm. The values of the min and max points when the q^{th} hyperbox is adjusted for the first time by using the $I_h = \left[I_h^l, I_h^u\right]$ are given by

$$V_q = I_h^l \quad , \quad W_q = I_h^u \tag{4}$$

   These values are similar to the input pattern.

2. *Hyperbox Expansion*:
   Suppose the h^{th} input pattern has to be expanded with the hyperbox B_q which have the highest degree of membership; before expansion the following condition has to be satisfied.

$$\forall_{a=1..n} \left(max\left(\left(w_{qa}, I_{ha}^u \right) - min\left(v_{qa}, I_{ha}^l \right) \right) \right) \leq \vartheta \tag{5}$$

   If this condition got satisfied, the new min and max points of the hyperbox B_q are given by

$$v_{qp}^{new} = min\left(v_{qp}^{old}, I_{hp}^l \right) \tag{6}$$

$$w_{qp}^{new} = max\left(w_{qp}^{old}, I_{hp}^u \right) \tag{7}$$

   And there is a case, when the above expansion condition does not satisfy, then we look for the other hyperboxes of the same class for the expansion. If neither hyperbox is ready for the expansion then make a new hyperbox B_k for the input pattern.

3. *Hyperbox Overlap test*:
   After the successful expansion, there are chances of overlap between the two hyperboxes and if both these hyperboxes belongs to the different classes then the classifier will give wrong results. The algorithm conducted the hyperbox overlap test to check for the overlap.
   Let the hyperbox B_q be expanded; we test for the overlap with hyperbox B_p if

$$class\left(B_q\right) = \begin{Bmatrix} 0, & test\ with\ all\ the \\ & other\ hyperboxes. \\ other, & go\ for\ the\ overlapping \\ & test\ only\ if \\ & class\left(B_q\right) \neq class(B_p) \end{Bmatrix} \tag{8}$$

   If suppose p^{th} dimension is detected from all the observations, and $\delta^{old} - \delta^{new} > 0$, then we set $\Delta = p$.

4. *Hyperbox Contraction*:
   Δ^{th} dimension of the two hyperboxes is adjusted only if $\Delta > 0$. To make minimal effect on the size and shape of the hyperbox only one dimension is adjusted in each hyperbox. The contraction phase of GFMM is very similar to the original Fuzzy min-max.

2.4 Network Architecture of GFMM

There are only two changes between the GFMM network architecture shown in Fig. 1 and Simpson's original FMM network architecture. First, the input node gets doubled to $2 * n$. Second, in the output layer, an additional node is introduced which handles the unlabeled hyperbox from the second layer of the network.

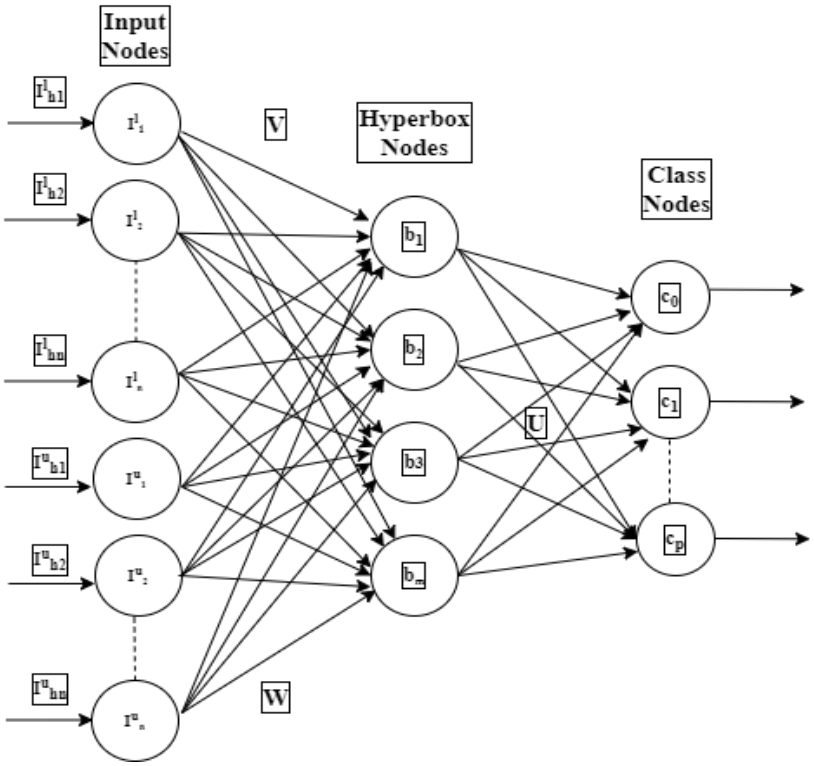

Fig 1. The network of GFMM

3 Enhanced Fuzzy Min-Max Neural Network

The Enhanced Fuzzy Min-Max Neural Network (EFMM) [4] overcomes the limitation of the original FMM learning algorithm and enhanced its performance. There are three heuristic rules for the learning algorithm, as will be discussed in this section.

3.1 Shortcomings of FMM

The three shortcomings of FMM that are overcome by EFMM, which makes an impact on the learning algorithm, are summarized below.

1. Hyperbox Expansion: In this phase they show that, when the overlapping regions are increasing between two classes it makes impact on the performance of the FMM. In FMM they first calculate the sum of all the differences between min and max point of the dimensions and then they compare this sum with the $n\vartheta$. There are very high chances of wrong prediction even if one dimension can exceed the $n\vartheta$ (expansion coefficient) and the sum of all dimension is under the expansion coefficient.

2. Hyperbox Overlap Test: The four existing cases for detecting the overlap between two different class hyperboxes are not sufficient. There are some inputs in which overlapping regions are detected and the test assumes it is a non-overlapping region and it stops the overlap test. So more conditions are added in the overlap test of EFMM.

3. Hyperbox Contraction: In FMM the contraction is based on the hyperbox overlap test, but the overlap test phase can pass some undetected overlapping regions which creates problems in the contraction phase.

In EFMM they modified all these three phases to overcome the problems. And the modified version improves the classification results.

3.2 EFMM Learning Algorithm

The three heuristic rules which can overcome all the limitations of EFMM are:

1. Hyperbox Expansion Rule: To solve all expansion problems in FMM, a new equation is formulated. The q^{th} hyperbox is checked from all the dimensions separately to see if it exceeds ϑ or not. This rule is only applicable if no dimension can exceed ϑ.

$$Max_n(W_{qp}, I_{hp}) - Min_n(V_{qp}, I_{hp}) \leq \vartheta \quad (9)$$

2. Hyperbox Overlap Test Rule: In the original FMM, the four cases are insufficient for the hyperbox overlap test. In GFMM, they modified the test phase and included additional overlap testing cases, as observed from (8). Now there are total nine cases to detect the possible overlap regions. And (10) and (11) are already there in FMM.

   Initially, $\delta^{old} = 1$

$$case\ 1: V_{qp} < V_{rp} < W_{qp} < W_{rp}\ ,\ \delta^{new} = min\left(W_{qp} - V_{rp}\ , \delta^{old}\right) \quad (10)$$

$$case\ 2: V_{rp} < V_{qp} < W_{rp} < W_{qp}\ ,\ \delta^{new} = min\left(W_{rp} - V_{qp}\ , \delta^{old}\right) \quad (11)$$

$$case\ 3: V_{qp} = V_{rp} < W_{qp} < W_{rp}\ ,\ \delta^{new} = min\left(min\left(W_{qp} - V_{rp}\ , W_{rp} - V_{qp}\ \right), \delta^{old}\right) \quad (12)$$

$$case\ 4: V_{qp} < V_{rp} < W_{qp} = W_{rp}\ ,\ \delta^{new} = min\left(min\left(W_{qp} - V_{rp}\ , W_{rp} - V_{qp}\ \right), \delta^{old}\right) \quad (13)$$

$$case\ 5: V_{rp} = V_{qp} < W_{rp} < W_{qp}\ ,\ \delta^{new} = min\left(min\left(W_{qp} - V_{rp}\ , W_{rp} - V_{qp}\ \right), \delta^{old}\right) \quad (14)$$

$$case\ 6: V_{rp} < V_{qp} < W_{rp} = W_{qp}\ ,\ \delta^{new} = min\left(min\left(W_{qp} - V_{rp}\ , W_{rp} - V_{qp}\ \right), \delta^{old}\right) \quad (15)$$

$$case\ 7: V_{qp} < V_{rp} \leq W_{rp} < W_{qp}\ ,\ \delta^{new} = min\left(min\left(W_{qp} - V_{rp}\ , W_{rp} - V_{qp}\ \right), \delta^{old}\right) \quad (16)$$

$$case\ 8: V_{rp} < V_{qp} \leq W_{qp} < W_{rp}\ ,\ \delta^{new} = min\left(min\left(W_{qp} - V_{rp}\ , W_{rp} - V_{qp}\ \right), \delta^{old}\right) \quad (17)$$

$$case\ 9: V_{rp} = V_{qp} < W_{rp} = W_{qp}\ ,\ \delta^{new} = min\left(\left(W_{rp} - V_{qp}\ \right), \delta^{old}\right) \quad (18)$$

   When $\delta^{old} - \delta^{new} < 1$, then only the overlapping region is detected. To check for the next dimension, we have to initialize $\Delta = p\ and\ \delta^{old} = \delta^{new}$. And this loop ends when no more regions are detected.

3. Hyperbox Contraction Rule:

   For the contraction of the overlapping hyperboxes, EFMM introduces nine cases and, all these cases are totally based on the overlap test rules.

$$Case\ 1: V_{q\Delta} < V_{r\Delta} < W_{q\Delta} < W_{r\Delta}, W_{q\Delta}^{new} = V_{r\Delta}^{new} = \frac{W_{q\Delta}^{old} + V_{r\Delta}^{old}}{2} \quad (19)$$

$$Case\ 2: V_{r\Delta} < V_{q\Delta} < W_{r\Delta} < W_{q\Delta}, W_{r\Delta}^{new} = V_{q\Delta}^{new} = \frac{W_{r\Delta}^{old} + V_{q\Delta}^{old}}{2} \quad (20)$$

$$Case\ 3: V_{q\Delta} = V_{r\Delta} < W_{q\Delta} < W_{r\Delta}, V_{r\Delta}^{new} = W_{q\Delta}^{new} \quad (21)$$

$$Case\ 4: V_{q\Delta} < V_{r\Delta} < W_{q\Delta} = W_{r\Delta}, W_{q\Delta}^{new} = V_{r\Delta}^{new} \quad (22)$$

$$Case\ 5: V_{r\Delta} = V_{q\Delta} < W_{r\Delta} < W_{q\Delta}, V_{q\Delta}^{new} = W_{r\Delta}^{new} \quad (23)$$

$$Case\ 6: V_{r\Delta} < V_{q\Delta} < W_{r\Delta} = W_{q\Delta}, W_{r\Delta}^{new} = V_{q\Delta}^{new} \quad (24)$$

$$Case\ 7(a): V_{q\Delta} < V_{r\Delta} \leq W_{r\Delta} < W_{q\Delta}\ and\ (W_{r\Delta} - V_{q\Delta}) < (W_{q\Delta} - V_{r\Delta}), V_{q\Delta}^{new} = W_{r\Delta}^{new} \quad (25)$$

$$Case\ 7(b): V_{q\Delta} < V_{r\Delta} \leq W_{r\Delta} < W_{q\Delta}\ and\ (W_{r\Delta} - V_{q\Delta}) > (W_{q\Delta} - V_{r\Delta}), W_{q\Delta}^{new} = V_{r\Delta}^{new} \quad (26)$$

$$Case\ 8(a): V_{r\Delta} < V_{q\Delta} \leq W_{q\Delta} < W_{r\Delta}\ and\ (W_{r\Delta} - V_{q\Delta}) < (W_{q\Delta} - V_{r\Delta}), W_{r\Delta}^{new} = V_{q\Delta}^{new} \quad (27)$$

$$Case\ 8(b): V_{r\Delta} < V_{q\Delta} \leq W_{q\Delta} < W_{r\Delta}\ and\ (W_{r\Delta} - V_{q\Delta}) > (W_{q\Delta} - V_{r\Delta}), V_{r\Delta}^{new} = W_{q\Delta}^{new} \quad (28)$$

$$Case\ 9(a): V_{q\Delta} = V_{r\Delta} < W_{q\Delta} = W_{r\Delta}, W_{q\Delta}^{new} = V_{r\Delta}^{new} = \frac{W_{q\Delta}^{old} + V_{r\Delta}^{old}}{2} \quad (29)$$

$$Case\ 9(b): V_{r\Delta} = V_{q\Delta} < W_{r\Delta} = W_{q\Delta}, W_{r\Delta}^{new} = V_{q\Delta}^{new} = \frac{W_{r\Delta}^{old} + V_{q\Delta}^{old}}{2} \quad (30)$$

These are the nine cases for the hyperbox contraction.

These three heuristic rules are the main reason for the enhancement of the learning algorithm of EFMM over FMM.

4 Methodology

The task at hand is to identify cancer subtypes from the gene expression profiles pertaining to lung cancer data. The details of the dataset are given in section 5. The

process flow of the training and testing processes for (GFMM / EFMM) is shown in Fig. 2. For each classifier, there are two different accuracies, one is with the selected genes and the second is with the original dataset. LASSO feature selection is used to select the significant genes prior to the classification phase. With the selected genes the whole classification process becomes faster and the results are also impressive.

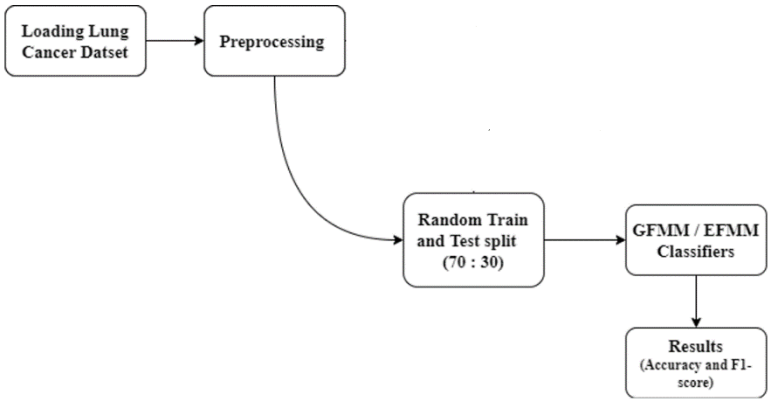

Fig 2. Flow chart of the training and testing process

The steps of the methodology are detailed below.

1. Load the Microarray gene expression dataset.
2. Normalize the dataset using min-max normalization and the range of the normalization is [0, 1]. In min-max normalization, all the minimum values are set to 0 and all the maximum values are set to 1. And the values which lie between maximum and minimum values are set with a decimal value within a range of [0, 1].
3. In this step we have two choices first is to go for classification without any feature selection and the second one is before going for the classification process select important features with LASSO and then perform the classification task.
    i. Directly go for train-test split and then perform classification by GFMM / EFMM.
    ii. Extract the important features from the lung cancer dataset. There is a requirement for this step because our dataset has 12600 genes (i.e. features) and not all the genes make an impact on the final result. We used the LASSO feature extraction technique for extracting the features. After the feature extraction process 176 genes got selected and these selected genes were used

for training of the model. This step makes the whole classification process faster and more efficient.

4. Divide the dataset into training and testing sets. 70:30 is the ratio we choose for the train:test split.
5. Now train the GFMM / EFMM model with the training set. For analyzing the performance of both the classifiers, calculate the accuracy and store them in their respective arrays. Repeat steps 4 and 5 for five times because we are performing a five-fold cross-validation.

5 Results

In this section we first discuss the experimental setup and the hyper parameter settings of the classification algorithms including GFMM and EFMM. Then we compare the results of GFMM and EFMM with the other classification algorithms.

5.1 Experimental Setup

The hyperparameters which are used for the various learning algorithms in this paper are given in Table 1. For some, the standard classification algorithms we used the default parameters, but for GFMM and EFMM we set the parameters to get the best result using grid search. All the experiments were performed in the python version 3.7.0 on Intel 2.00GHz core PC.

Table 1. Hyperparameter Settings

CLASSIFIER	HYPER PARAMETER	VALUES
Enhanced Fuzzy Min-Max	Hyperbox Coefficient(ϑ)	0.1
	Sensitivity(ω)	1
General Fuzzy Min-Max	Hyperbox Coefficient(ϑ)	0.1
	Sensitivity(ω)	1
Fuzzy Min-Max	Hyperbox Coefficient(ϑ)	0.7
	Sensitivity(ω)	1

Support Vector Machine	Regularization parameter (c)	1
	Gamma	0.0018, 0.126 (with LASSO)
K-Nearest Neighbor	No. of neighbors	7
Logistic Regression	C	1
	Penalty	l2
	Solver	lbfgs
Naïve Bayes	Var_Smoothing	$1e^{-9}$
Random Forest	N_estimators	100
	Max_depth	2

We used microarray lung cancer dataset [20] for all the experiments. This dataset has 203 samples and 12,600 features (genes). There are five classes indicating five subtypes of lung cancer [20]. The class distribution is highly imbalanced. The different cancer subtypes and their class populations are: lung adenocarcinomas (139), squamous cell lung carcinomas (21), lung carcinoids (20), small cell lung carcinomas (6), and normal samples (17). Under such a scenario, defining accurate class boundaries is an obvious challenge. We propose to counter this challenge using the FMM classifiers: GFMM and EFMM.

5.2 GFMM Results

The General Fuzzy min-max model hyperbox visualization after training is complete is shown in Fig. 3. The five colors indicate the five classes. The hyperbox visualization shows some degree of overlap between a few classes, especially between the majority class and two of the minority classes. For the GFMM algorithm, the value we choose for the hyperbox expansion coefficient (i.e. theta) is 0.1 and the sensitivity value is 1; sensitivity measures the fuzziness and this is used in the testing phase. The classification results are shown in Tables 2 (accuracy) and 3 (execution time) for all methods. For GFMM, we observe that the execution time of the classification process in case of selected features is 4.57 secs which is faster as compared to all the other algorithms. The accuracy we achieved with and without LASSO are 95.41% and 89% respectively.

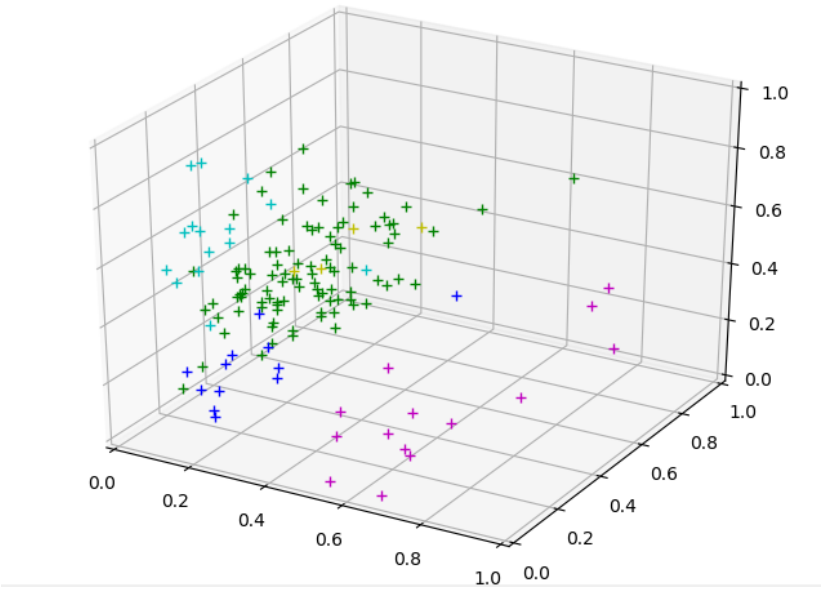

Fig 3. GFMM hyperbox after training on the lung cancer gene expression data

5.3 EFMM Results

Enhanced Fuzzy min-max classifier gives the best result among all the classifiers, as observed from Table 2. The accuracy achieved with and without LASSO is 97.37% and 91.80% respectively, and this is the best among all the classifiers that we have used for this microarray dataset. The EFMM hyperbox visualization is shown in Fig. 4, that is obtained after the training process is complete. Comparing the hyperbox visualizations of GFMM in Fig. 3 with that of EFMM in Fig. 4 we observe a better segregation of classes in case of EFMM in Fig. 4, indicating that the fuzzy membership functions for the five classes are more well defined in case of EFMM. The reason for the more accurately defined class boundaries of EFMM is the improved learning process aiming to reduce the overlap between hyperboxes. As observed from the hyperbox visualization in Fig. 4, the majority class namely, lung adenocarcinoma, is segregated well from the other classes which improves the classification performance.

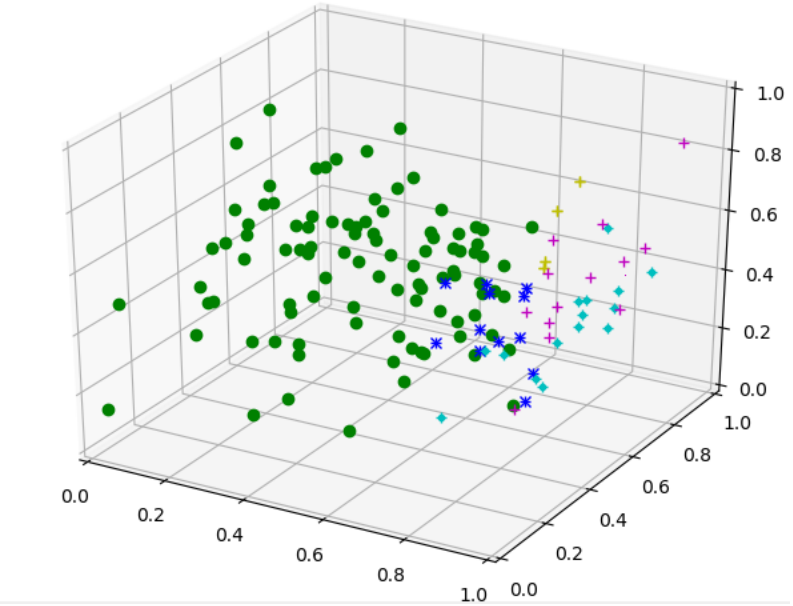

Fig 4. EFMM Hyperbox after training on the lung cancer gene expression data

5.4 Results Comparisons

In Table 2 we can see accuracy comparison of all the classification algorithms. Other than FMM, we have compared the results to Support Vector Machine (SVM), K-Nearest Neighbor, Logistic Regression, Naïve Bayes and Random Forest. From these results, we analyze that EFMM stands out among all the algorithms in terms of the accuracy obtained with LASSO feature selection. Without LASSO, SVM gives the best result followed by EFMM. The GFMM is also better in terms of execution time but the accuracy is almost same as the Simpson`s Fuzzy min-max classifier.

Table 2. Accuracies for lung cancer classification

Algorithms	**Accuracy (in %)**	
	With LASSO	**Without LASSO**
Enhanced Fuzzy Min-Max	97.37	91.80
General Fuzzy Min-Max	95.41	89.00
Fuzzy Min-Max	95.08	89.5
Support Vector Machine (SVM)	94.09	92.45
K-Nearest Neighbor	93.77	89.18
Logistic Regression	92.13	90.18
Naïve Bayes	92.78	89.18

Random Forest	83.31	80.32

From the execution times listed in Table 3, we observe that FMM based methods take more time to execute as compared to the other machine learning algorithms. GFMM is fastest in case of selected features compare to other fuzzy algorithms, and EFMM takes approximately same time for both with and without selected features.
After analyzing all the comparison, we can say that the Enhanced fuzzy min-max classifier is a very suitable option for the classification of microarray data, and it performs best when used with feature selection techniques for selecting the most optimal gene set that identifies the cancer subtype well.

Table 3. Execution time of classification algorithms

Algorithms	**Execution Time (in seconds)**	
	With LASSO	**Without LASSO**
Enhanced Fuzzy Min-Max	40.21	41.54
General Fuzzy Min-Max	4.57	87.52
Fuzzy Min-Max	13.77	963.31
Support Vector Machine (SVM)	0.40	17.74
K-Nearest Neighbor	0.37	0.37
Logistic Regression	0.66	8.51
Naive Bayes	0.32	1.39
Random Forest	1.98	2.90

6 Conclusion

In this work, we explore the generalized and enhanced versions of the fuzzy min-max neural network for application on microarray gene expression data. For all the experiments we used the microarray lung cancer dataset. LASSO is used for selecting the important genes and this optimized subset of genes is used in the training of GFMM and EFMM. In the performance analysis of both the classifiers, we found that EFMM is more efficient as compared to GFMM and Simpson`s FMM in terms of both accuracy and execution time. EFMM also outperforms all other machine learning algorithms when used in combination with feature selection. The hyperbox visualizations indicate that the fuzzy membership values for the hyperboxes pertaining to the five lung cancer subtypes are better defined in case of EFMM that for GFMM. The decision boundaries in case of EFMM are more accurate due to the improved expansion-contraction process that aims to reduce the overlap between the hyperboxes representing the different classes. GFMM is the fastest among all the fuzzy classifiers but the classification performance of GFMM is same as the Simpson`s FMM. In our study, we prove that EFMM in combination with LASSO is the most effective technique for classifying the high-

dimensional small sample gene expression data. Exploring hybrid combinations of EFMM with machine learning methods is the future scope of this work.

Conflict of Interest

The authors declare no potential conflict of interest.

References

1. Simpson, Patrick K. "Fuzzy Min—MaX Neural NetWorks—Part 1: Classification." *IEEE Trans. on Neural Networks* 3, no. 5 (1992): 776-786.
2. Khuat, Thanh Tung, Dymitr Ruta, and Bogdan Gabrys. "Hyperbox-based machine learning algorithms: a comprehensive survey." *Soft Computing* 25, no. 2 (2021): 1325-1363.
3. Gabrys, Bogdan, and Andrzej Bargiela. "General fuzzy min-max neural network for clustering and classification." *IEEE transactions on neural networks* 11, no. 3 (2000): 769-783.
4. Mohammed, Mohammed Falah, and Chee Peng Lim. "An enhanced fuzzy min–max neural network for pattern classification." *IEEE transactions on neural networks and learning systems* 26, no. 3 (2014): 417-429.
5. Singh, Yashpal, and Seba Susan. "Optimal Gene Selection and Classification of Microarray Data Using Fuzzy Min-Max Neural Network with LASSO." In *International Conference on Intelligent and Fuzzy Systems*, pp. 777-784. Springer, Cham, 2022.
6. Tibshirani, Robert. "Regression shrinkage and selection via the lasso." *Journal of the Royal Statistical Society: Series B (Methodological)* 58, no. 1 (1996): 267-288.
7. Susan, Seba, and Madasu Hanmandlu. "Smaller feature subset selection for real-world datasets using a new mutual information with Gaussian gain." *Multidimensional Systems and Signal Processing* 30, no. 3 (2019): 1469-1488.
8. Slonim, Donna K., Pablo Tamayo, Jill P. Mesirov, Todd R. Golub, and Eric S. Lander. "Class prediction and discovery using gene expression data." In *Proceedings of the fourth annual international conference on Computational molecular biology*, pp. 263-272. 2000.
9. Khan, Javed, Jun S. Wei, Markus Ringner, Lao H. Saal, Marc Ladanyi, Frank Westermann, Frank Berthold et al. "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks." *Nature medicine* 7, no. 6 (2001): 673-679.
10. Lyu, Boyu, and Anamul Haque. "Deep learning based tumor type classification using gene expression data." In *Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics*, pp. 89-96. 2018.
11. Ahmed, Omar, and Adnan Brifcani. "Gene expression classification based on deep learning." In *2019 4th Scientific International Conference Najaf (SICN)*, pp. 145-149. IEEE, 2019.
12. Brown, Michael PS, William Noble Grundy, David Lin, Nello Cristianini, Charles Walsh Sugnet, Terrence S. Furey, Manuel Ares Jr, and David Haussler. "Knowledge-based analysis of microarray gene expression data by using support vector machines." *Proceedings of the National Academy of Sciences* 97, no. 1 (2000): 262-267.
13. Ram, Malihe, Ali Najafi, and Mohammad Taghi Shakeri. "Classification and biomarker genes selection for cancer gene expression data using random forest." *Iranian journal of pathology* 12, no. 4 (2017): 339.

14. Sartor, Maureen A., George D. Leikauf, and Mario Medvedovic. "LRpath: a logistic regression approach for identifying enriched biological groups in gene expression data." *Bioinformatics* 25, no. 2 (2009): 211-217.
15. Chandra, B., and Manish Gupta. "Robust approach for estimating probabilities in Naïve–Bayes Classifier for gene expression data." *Expert Systems with Applications* 38, no. 3 (2011): 1293-1298.
16. Kang, Chuanze, Yanhao Huo, Lihui Xin, Baoguang Tian, and Bin Yu. "Feature selection and tumor classification for microarray data using relaxed Lasso and generalized multi-class support vector machine." *Journal of theoretical biology* 463 (2019): 77-91.
17. Pedrycz, Witold. "Fuzzy sets in pattern recognition: methodology and methods." *Pattern recognition* 23, no. 1-2 (1990): 121-146.
18. Carpenter, Gail A., Stephen Grossberg, Natalya Markuzon, John H. Reynolds, and David B. Rosen. "Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps." *IEEE Transactions on neural networks* 3, no. 5 (1992): 698-713.
19. Pedrycz, Witold. "Fuzzy neural networks with reference neurons as pattern classifiers." *IEEE Transactions on Neural Networks* 3, no. 5 (1992): 770-775.
20. Zhu, Zexuan, Yew-Soon Ong, and Manoranjan Dash. "Markov blanket-embedded genetic algorithm for gene selection." *Pattern Recognition* 40, no. 11 (2007): 3236-3248.